5-O

УДК 681.03.06

***А.И. Ольшевский***
Государственный университет информатики и искусственного интеллекта,
г. Донецк, Украина
info@iai.donetsk.ua

# Разработка алгоритмов генерации имитационных моделей коммуникационных сетей дистанционного обучения

Статья посвящена вопросам автоматизации имитационного моделирования при проектировании сетей дистанционного обучения. Для реализации алгоритмов генерации имитационных моделей на GPSS предложена декомпозиция модели коммуникационной сети дистанционного обучения. Также приведены: представление отдельных частей модели, алгоритмы работы и интерпретация алгоритмов на языке моделирования в среде GPSS World.

## Введение

Первая и, возможно, наиболее значимая проблема для системы образования будущего – это интеграция, создание единого образовательного и информационного пространства. Это – стратегическая проблема, направленная на перспективное развитие систем образования разных стран мира, которые осознают единство и целостность мира, взаимосвязь и взаимообусловленность его составных частей.

Особую роль в аспекте единого образовательного пространства могут и должны сыграть новые информационные технологии. В данном случае хотелось бы остановиться на возможностях, которые несут в себе глобальная сеть Интернет и информационные системы дистанционного обучения. Студенты имеют доступ к разнообразным информационным банкам данных по всем странам мира, могут работать вместе над проектом, который их интересует, обсуждать проблемы практически со всем миром. Человек, в самом деле, в любой период своей жизни приобретает возможность дистанционно получить новую профессию, повысить свою квалификацию, расширить свое мировоззрение, причем практически в любом научном или учебном центре мира.

Решение задач создания сетей обучения может быть обеспечено специальной системой, ориентированной на предварительное исследование информационных потоков предметной области и стоимостных зависимостей, синтеза топологии компьютерных сетей и решения задач маршрутизации. Такая система создается для организации дистанционного обучения на базе ГУИиИИ: Государственного университета информатики и искусственного интеллекта (Донецк) [1].

Необходимой частью такой системы является имитационное моделирование сети дистанционного обучения для оценки ее функционирования.

Программные средства моделирования включают в себя различные имитационные модели сети обучения в соответствии с ее структурой и функционированием [2].

Программное обеспечение автоматизации имитационного моделирования разрабатывалась таким образом, чтобы построение моделей логически следовало за процессом оптимизации структуры системы. Однако процесс имитационного моделирования стано-

вится неоправданно трудоемким при изменении входных данных или структуры сопряжения элементов системы. В процессе варьирования различных вариантов структуры сети обучения генерируются имитационные модели на GPSS.

С этой точки зрения весьма эффективной и целесообразной является автоматизация построения имитационных моделей, способная быстро настраиваться на объекты любой структуры и вида и, тем самым, коренным образом уменьшать время подготовки моделей и внесение изменений.

Имитационное моделирование является наиболее универсальным методом исследования систем и количественной оценки характеристик их функционирования. При создании имитационных моделей возникают задачи организации псевдопараллельного выполнения алгоритмов; операций с модельным временем; имитации случайных процессов и т.п. Программа создается автоматически по одной из формализованных схем на основании задаваемых пользователем параметров системы, особенностей функционирования [3]. Введение имитационного моделирования позволяет принимать решения о построении сетей без проведения дорогостоящего натурного моделирования и дает возможность существенно удешевить процесс создании сетей дистанционного обучения.

Таким образом, целью статьи является исследование коммуникационной сети и разработка алгоритмов генерации имитационной модели сети дистанционного обучения в соответствии с ее топологией и характеристиками.

# Декомпозиция модели коммуникационной сети системы дистанционного обучения

Построение модели сложной системы представляет собой довольно сложный, трудоемкий процесс. От точности определения функциональных взаимосвязей между отдельными фрагментами реальной системы, отражаемой моделью, а также от уровня детализации каждого из этих фрагментов, напрямую зависит и точность полученных результатов моделирования, уровень достоверности этих результатов. Поэтому, при построении модели системы любой сложности, особое внимание следует уделить этапу декомпозиции самой системы – как начальному этапу проектирования.

Что касается декомпозиции, задачи построения модели коммуникационной сети системы дистанционного обучения, то здесь на первый план выходит выбор точки зрения на рассмотрение всей системы в целом. Выбор точки зрения определит список конечных, простейших задач, решения которых в совокупности и будут представлять собой решение обобщенной задачи. Выбранная точка зрения – представление системы, прежде всего как коммуникационной сети, осуществляющей обмен информацией между конечными узлами, а уже потом как системы дистанционного обучения, являющейся методом и средством познания и обучения для ее пользователей, наиболее точно отражает сущность данной работы.

На первом шаге детализации происходит упрощение модели системы, рассматриваемой в качестве сети для обмена информацией. Уже на этом уровне вся система представляет собой систему массового обслуживания, где в качестве источника заявок выступает все множество конечных узлов (ЭВМ-«участников» сети), в качестве обслуживающего устройства – сама коммуникационная сеть, в качестве транзактов – пакеты, исходящие от компьютеров пользователей сети [4]. Кроме того, сами конечные узлы также являются приемниками и обработчиками транзактов.

# Представление модельного времени

Для представления времени в модели необходимо учесть минимальное время, с которым изменяется хотя бы один значимый процесс (слово «значимый» означает, что изменяемые параметры, атрибуты либо характеристики процесса влияют на другие процессы либо на модель в целом, т.е. служат входными параметрами для них). После определения такого времени частота дискретизации модельного времени выбирается таким образом, чтобы порядок единицы модельного времени был равным или близким порядку времени, за которое изменяется процесс с минимальной скоростью изменения.

В случае моделирования работы коммуникационной сети был проанализирован ряд характеристик основных элементов сети и принято решение, что оптимальным выбором единицы модельного времени для моделирования работы коммуникационной сети является $10^{-6}$ с, или 1 микросекунда (мкс), либо близкое к этому значение $10^{-5}$ с.

Для удобного перевода реального времени в модельное удобно воспользоваться некоторым множителем, применяемым ко всем временным величинам в модели, отражающим поведение некоторого ее элемента. Назовем эту величину множителем модельного времени (ММВ):

$$MMB = \frac{1}{t_d},$$

где $t_d$ – единица модельного времени.

Таким образом, для перевода реального времени в модельное достаточно умножить значение реального времени на ММВ, а для представления результатов в удобном для пользователя виде – разделить результат на ММВ. Это дает возможность изменять частоту дискретизации модельного времени с целью повысить быстродействие без существенного изменения всей модели.

# Модель канала связи

***Время передачи сообщения по каналу связи.*** Любая физическая среда, предназначенная для передачи информации (будь то медный кабель, оптоволокно или радиоэфир), вносит определенную задержку при передаче по нему сигнала, представляющего некоторый объем информации. Эта задержка зависит от объема информации, от расстояния между передатчиком и приемником (т.е. от протяженности линии связи), а также от пропускной способности линии связи.

Скорость распространения электричества по проводнику принято считать равной скорости света в вакууме (или близкой к ней), равной 300 000 000 м/с. То есть время прохождения одного бита информации по линии связи протяженностью $L$ м будет равно:

$$T_{1б} = \frac{L}{V_{св}},$$

где $T_{1б}$ – время передачи одного бита информации по проводнику длиной $L$;

$V_{св}$ – скорость света в вакууме, равная скорости распространения электричества.

Однако для определения задержки времени пересылки электрического сигнала от одного конца проводника к другому недостаточно одного отношения длины проводника к скорости распространения сигнала. Ведь информация передается кадрами, в которых отдельные биты или последовательности битов кодированы согласно определенному алгоритму в электрический сигнал различной частоты и амплитуды (в зависимости от

алгоритма кодирования). Следовательно, общее время пересылки одного кадра равняется времени, прошедшему между отправкой передатчиком первого бита информации и получением приемником последнего бита. Данная величина рассчитывается путем сложения времени $T_{1б}$ с величиной задержки передачи данных, которая представляет собой время между отправкой передатчиком первого и последнего бита и зависит от пропускной способности канала (от скорости передачи данных) следующим образом:

$$t_{зпд} = \frac{V_к}{C},$$

где $t_{зпд}$ – время задержки передачи сигнала; $V_к$ – объем кадра в битах; С – скорость передачи данных.

Исходя из вышесказанного, время передачи по линии связи одного кадра вычисляется по формуле:

$$t_{1к} = \frac{L}{V_{св}} + \frac{V_к}{C},$$

где $t_{1к}$ – время передачи одного кадра размером $V_к$ бит по каналу связи протяженностью $L$ м со скоростью $C$ бит/с.

Данная величина будет получена в секундах. Однако для использования в модели ее удобно выразить в единицах модельного времени при помощи ММВ:

$$t_{1к} = (\frac{L}{V_{св}} + \frac{V_к}{C}) * ММВ \tag{1}$$

**Имитация передачи сообщения каналом связи.** Линия связи всего лишь вносит задержку передачи некоторого объема информации между двумя узлами. Причем количество одновременно передаваемых кадров одной линией неограниченно. Кадры поступают в физическую среду с определенным шагом и движутся к приемнику независимо друг от друга (параллельно). Таким образом, транзакт в модели, имитирующий кадр, движется от одного устройства к другому (от передатчика к приемнику), достаточно на его пути поставить блок задержки продвижения транзакта, и параметру, отвечающему за время задержки, присвоить значение $t_{1к}$, вычисленное по формуле (1).

Блок ADVANCE задерживает каждый транзакт на время $t_{1к}$. Второй параметр определяет модификатор времени задержки транзактов, равный нулю в данном случае, так как все транзакты задерживаются на точно определенный промежуток времени.

Например, пусть требуется сымитировать передачу пакета размером 1,5 Кб физической средой протяженностью 100 м. Кадр передается со скоростью 100 Мб/с. Фрагмент модели на языке GPSS приведен ниже:

```
LenChan1 EQU 100 ;длина линии связи
WaveSpeed EQU 300000000 ;скорость прохождения сигнала (скорость света)
Lan1Speed EQU 100000000 ;скорость передачи данных
VFrame1 EQU 1500 ;размер кадра в байтах
DisFreq EQU 1000000 ;MMB
Time1 VARIABLE (LenChan1/WaveSpeed+8#VFrame1/Lan1Speed)#DisFreq
;арифметическая переменная, общая задержка передачи кадра звеном
... ;блоки, имитирующие сетевое устройство передатчика
ADVANCE V$Time1 ;передача кадра (задержка)
... ;блоки, имитирующие сетевое устройство приемника
```

# Маршрутизация

Для продвижения пакетов по сети от узла-источника к узлу-приемнику используются таблицы маршрутизации, которые строятся интеллектуальными средствами сетевых коммуникационных устройств в процессе самообучения, занимающего небольшой промежуток времени, в ходе которого происходит обмен сообщениями между оборудованием сети с целью построения обобщенного представления о структуре сети: о ее архитектуре и топологии, о ближайших соседях, о конечных узлах и приблизительном их местонахождении относительно собственного расположения.

Обобщенно таблица маршрутизации для коммутатора S1 некоторой абстрактной сети может иметь следующий вид (табл. 1):

Таблица 1 – Таблица маршрутизации коммутатора S1

| **Адрес назначения** | **Адрес следующего коммутатора** |
|---|---|
| N1 | Пакет не требуется передавать через сеть |
| N2 | S2 |
| N3 | S3 |
| N4 | S3 |
| N5 | S2/S3 |

Из приведенной таблицы маршрутизации можно сделать вывод, что узел N1 непосредственно примыкает к коммутатору S1 и будет следующим и последним этапом продвижения пакета, адресованного ему. Для передачи пакета узлу N2 необходимо вначале передать его коммутатору S2, тогда как для узлов N3 и N4 следующим транзитным коммутатором будет S3. Для узла N5 таблица маршрутизации содержит более одного коммутатора. Это позволяет устройству интеллектуально управлять маршрутом передачи пакета для наиболее быстрой его передачи к узлу назначения, используя альтернативные транзитные узлы при чрезмерной нагрузке основного (или выходе его из строя).

При моделировании работы коммутатора, в частности при моделировании его таблицы маршрутизации, всплывают две основные проблемы, зависящие от способа представления данных и алгоритма построения маршрута пакета:

1) если строить таблицу маршрутизации для каждого коммутатора сети, то это приведет к нерациональному использованию памяти, так как большая часть каждой таблицы будет дублировать большинство других, а для вычислительной сети средних масштабов с множеством коммутаторов и конечных станций отдельная таблица маршрутизации требует довольно больших объемов памяти;

2) алгоритмы построения маршрута для каждого пакета требуют некоторого числа итераций цикла, изменяющего параметры каждого транзакта. Учитывая тот факт, что одновременно в модели коммуникационной сети довольно большое множество транзактов являются активными, становится очевидным и факт увеличения реального времени моделирования системы из-за выполнения большого количества операций.

Решением первой проблемы стало построение единой унифицированной таблицы маршрутизации для всех коммутирующих устройств сети. Данная таблица

представляет собой матрицу размерностью N*N*M, где N – количество узлов (в том числе и конечных) графа, представляющего сеть, M – максимальное количество альтернативных путей между двумя несмежными вершинами. Построение таблиц маршрутизации для каждого коммутатора потребовало бы в среднем в два раза больше объема памяти, ведь тогда требовалось бы N матриц размерностью 2*N*M, причем каждая из них содержала бы часть информации других таблиц.

Вторая проблема также предполагает некоторую двоякость: было составлено два алгоритма построения маршрута отдельного транзакта. Причем в одном из них время моделирования сведено к минимуму, однако отсутствует возможность альтернативного выбора маршрута в связи с недоступностью основного. Во втором эта проблема была решена за счет увеличения общего количества операций по изменению параметров транзактов всей модели в целом. Выбор был сделан в пользу второго алгоритма, так как он более точно отражает процесс продвижения пакетов в сети из-за возможности выбора альтернативного транзитного пути.

Фрагмент модели, построенной в системе моделирования GPSS World, имитирующий первый алгоритм определения маршрута, приведен ниже:

```
MarTab MATRIX ,8,8 ;инициализация таблицы маршрутизации
INITIAL MX$MarTab(1,2),5
...
INITIAL MX$MarTab(8,7),7

GENERATE 1 ;генерация кадра
ASSIGN 1,4 ;параметр 1 - для косвенной адресации
ASSIGN 2,1 ;параметр 2 - метка узла-источника
ASSIGN 3,7 ;параметр 3 - метка узла-приемника

Iter ASSIGN *1,MX$MarTab(*2,*3) ;i+4-му параметру транзакта присваивается номер очередного узла
ASSIGN 2,MX$MarTab(*2,*3) ;второму параметру присваивается номер очередного узла
ASSIGN 1+,1 ;инкремент параметра косвенной адресации
TEST E P2,P3,Iter ;проверка условия, достигнут ли желаемый узел?
ASSIGN 1,4 ;восстановление первого параметра косвенной адресации для передачи пакета первому транзитному узлу
TRANSFER ,*1 ;безусловная передача пакета на канал связи к первому транзитному узлу, метка в параметре 4
```

При приеме транзакта очередным транзитным узлом осуществляется наращивание его первого параметра на единицу, обработка (задержка) и передача по метке, указанной в следующем параметре, адресованном первым параметром.

Во втором алгоритме решается проблема выбора альтернативного маршрута ценой увеличения времени работы модели.

Суть алгоритма состоит в определении следующего узла в маршруте пакета на каждом шаге его продвижения (а не заранее, как в предыдущем алгоритме). Каждый фрагмент модели, имитирующий работу отдельного узла коммутации, должен содержать следующий код, определяющий метку следующего узла (пример на языке GPSS):

```
...
;узел сети, помеченный первым
d1 ASSIGN 2,1 ;присвоение третьему параметру транзакта номера узла сети
TEST NE P3,P2,Term ;условие, достиг ли транзакт узла назначения?
```

```
TRANSFER ,MX$MarTab(*3,*2) ;если не достиг - переход на метку, указанную
"таблице маршрутизации" согласно параметрам 2 и 3 транзакта
...
Term ... действия с транзактом при условии достижения узла назначения
```

# Модель коммутатора

Назначение фрагмента модели, соответствующего работе коммутатора в составе коммуникационной сети – внесение задержки при продвижении транзактов, имитирующих отдельные пакеты, перемещаемые по сети, а также осуществление перенаправления каждого транзакта (маршрутизации).

Еще одна задача, возникающая при построении имитационной модели работы коммутатора – обработка транзактов различных приоритетов. Ведь современные коммутаторы способны различать приоритеты поступающих пакетов для достижения приемлемого уровня таких показателей качества связи, как скорость доставки и качество доставки.

В результате детального анализа предметной области удалось разделить все множество транзактов на две категории: срочные (приоритет 2) и несрочные (приоритет 1), в связи с этим алгоритмы приведены для обработки транзактов двух приоритетов с рекомендациями касательно изменения алгоритма (и структуры модели) при увеличении количества приоритетов.

Распределенный режим несколько отличается от приоритетного и имеет свои преимущества касательно применения в сетях с относительно равным количеством информации, передаваемой срочными и бессрочными потоками. Так, в алгоритм распределенного режима обработки добавился один параллельный блок, содержащий бесконечный цикл. В нем в самом начале алгоритма генерируется один транзакт, который затем циклически переходит от одного блока задержки на некоторое время $t_1$, характерное для времени обслуживания буфера пакетов приоритета 1 физического интерфейса, ко второму блоку задержки на время $t_2$, характерное для времени обслуживания буфера пакетов приоритета 2. Доступность же свободного процессора каждому буферу определяется текущим местоположением транзакта в блоке распределения.

Пример реализации блока распределения в среде моделирования GPSS World приведен далее:

```
M1I1Time EQU Time1 ;время обслуживания процессором кадра
M1I1Pr1Act EQU TimePart1 ;доля процессорного времени для приоритета 1
M1I1Pr2Act EQU TimePart2 ;доля процессорного времени для приоритета 2
M1I1Pr1Time VARIABLE (M1I1Time#M1I1Pr1Act#koef) ;процессорное время для
приоритета 1
M1I1Pr2Time VARIABLE (M1I1Time#M1I1Pr2Act#koef) ;процессорное время для
приоритета 2
GENERATE ,,,1 ;генерация одного транзакта при старте модели
LM1I1Act SEIZE M1I1Pr1Activity ;
ADVANCE V$M1I1Pr1Time ;очередь с приоритетом 1 активна
RELEASE M1I1Pr1Activity
SEIZE M1I1Pr2Activity
ADVANCE V$M1I1Pr2Time ;очередь с приоритетом 2 активна
RELEASE M1I1Pr2Activity
TRANSFER ,LM1I1Act ;бесконечный цикл с одним транзактом
```

В данном примере используются две константы M1I1Pr1Act и M1I1Pr2Act, указывающие процентное соотношение времени процессора для обработки каждой очереди. Необходимо учесть следующие ограничения, накладываемые на значения этих констант:

1) 0 < M1I1Pr1Act, M1I1Pr2Act < 1;

2) M1I1Pr1Act+M1I1Pr2Act = 1.

Следующие арифметические переменные M1I1Pr1Act, M1I1Pr2Act вычисляют конкретные значения времен задержек переключающего транзакта в каждом блоке задержки. Значение каждой такой переменной вычисляется по формуле:

$$t_i = T_{proc} * Tq_i * koef * MMB, \quad (3)$$

где $t_i$ – фактическое время обслуживания очереди $i$ процессором; $T_{proc}$ – время обслуживания процессором пакета; $Tq_i$ – доля времени обслуживания очереди i; *koef* – некоторый множитель, $koef > T_{proc}$; *MMB* – множитель модельного времени для представления значения в единицах модельного времени.

Следует отметить, что для моделирования работы всего коммутатора в распределенном режиме (а то и всех коммутаторов модели – при условии, что все они абсолютно идентичны) достаточно наличия одного блока распределения – при условии, что все физические интерфейсы имеют равное количество очередей, и каждой из них выделяется равная доля времени.

## Модель конечной станции

Первым этапом построения наиболее корректной модели конечной станции коммуникационной сети является ее формальное описание с точки зрения функционирования, т.е. описание функций, поддерживаемых данным объектом модели, а в связи с этим – описание типов и свойств информационных потоков (транзактов в имитационном моделировании), посредством которых осуществляется взаимодействие данного объекта модели с моделью в целом.

Прежде всего, удалось выделить два типа конечных узлов в сети дистанционного обучения: станции студентов и станции преподавателей. В зависимости от типа конечного узла был определен и ряд задач, свойственных каждому типу.

При моделировании логики работы конечных узлов коммуникационной сети разработчик сталкивается с рядом проблем, вызванных необходимостью уменьшения машинного времени работы модели как следствия большого количества блоков языка моделирования, в свою очередь, вызванного достаточной сложностью моделируемого объекта.

Так как модели конечных узлов в общей модели коммуникационной сети являются основными источниками транзактов, то именно в них и сосредоточено большое количество блоков генерации транзактов. Снижение их количества без ущерба для логики работы модели приведет к уменьшению машинного времени моделирования, чего и требуется добиться.

Идея разработанного алгоритма основана на классификации задач, выполняемых моделью конечного узла, по категории обязательности и регулярности. Так, для каждого класса транзактов используется свой блок генерации.

На рис. 1 приведена обобщенная схема разработанного алгоритма, а также его положение в модели конечной станции коммуникационной сети.

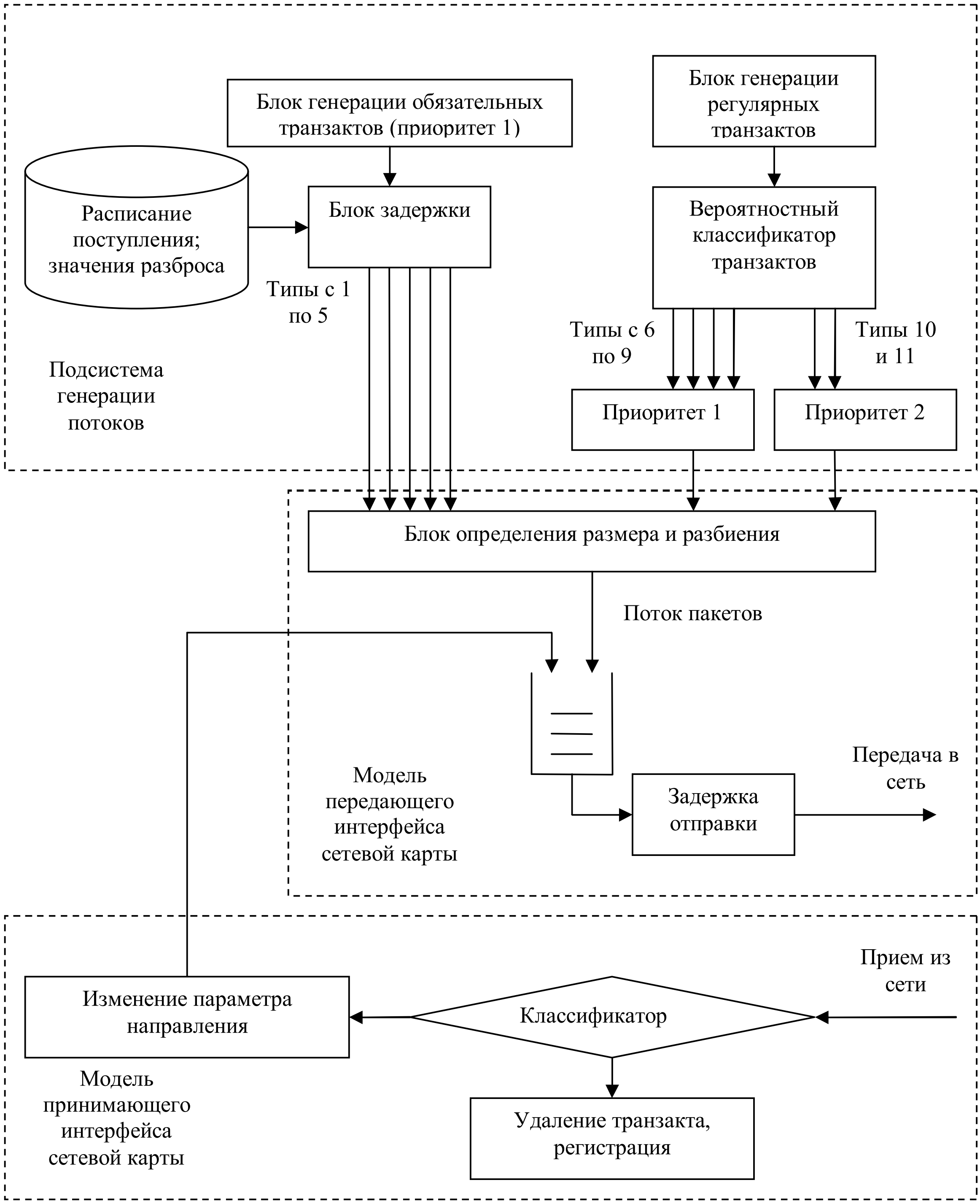



Рисунок 1 – Обобщенная схема функционирования модели станции студента

При реализации данного алгоритма на языке моделирования особый интерес представляет блок генерации обязательных транзактов и связанная с ним задача составления расписания времени их поступления в модель. Данная процедура

является полностью автоматизированной и возлагается на программу генерации модели конечного узла на основании учебного плана как исходных данных генерируемой модели.

## Модель передающего физического интерфейса сетевой карты

После генерации транзакта, имитирующего задание, присвоения ему типа и приоритета (а также иной предварительной инициализации) всегда осуществляется безусловный переход по некоторой метке к блоку, имитирующему передающий физический интерфейс сетевой карты. Назначение данного блока заключается в следующем:

1) определении количества транзактов, имитирующих поток информации, в соответствии с объемом, свойственным данному типу потока;
2) определении других параметров транзакта, свойственных данному типу потока (например, адресат – для электронных писем, идентификатор сервера, содержащего файловый архив и т.д.);
3) разбиении транзакта на определенное количество копий;
4) организации очереди транзактов, ожидающих передачи в сеть;
5) осуществлении маршрутизации;
6) задержке транзакта при выходе из блока, имитирующей работу сетевой карты;
7) передаче транзакта по метке к модели канала связи.

## Модель конечной станции преподавателя

Общая структура и логика функционирования станции преподавателя в целом повторяет структуру и логику функционирования станции студента. Небольшие различия касаются лишь блока генерации заданий (это связано с различиями в количестве заданий), а также фрагмента модели, имитирующего принимающий интерфейс (что связано с небольшими различиями в функциях некоторых заданий).

## Модель сервера

По сути сервер в коммуникационной сети также можно отнести к типу конечных узлов. Однако его назначение довольно специфическое, что во многом отличает его от рассмотренных типов конечных узлов: станций студентов и преподавателей. Так, сервер не генерирует транзактов, а алгоритмы обработки поступающих транзактов различных типов коренным образом отличаются от алгоритмов их обработки иными узлами модели сети.

В исследованной иерархической структуре коммуникационной сети системы дистанционного обучения сервер является единственным типом узлов, которые должны обрабатывать заявки всех выделенных типов. От точности определения хода обработки заявок каждого типа зависит степень адекватности модели сервера реальному объекту, выполняющему моделируемые процессы.

В табл. 2 приведены основные этапы обработки моделью сервера транзактов каждого типа.

Таблица 2 – Этапы обработки сервером транзактов каждого типа

| Тип транзакта | Порядок обработки |
|---|---|
| **Тип 1** (реферат);<br>**Тип 2** (контрольная работа);<br>**Тип 3** (лабораторная работа);<br>**Тип 4** (курсовая работа);<br>**Тип 7** (электронное письмо); | 1) принять все транзакты одного семейства;<br>2) объединить транзакты одного семейства;<br>3) изменить параметр назначения согласно параметру адресата;<br>4) поместить транзакт в модель почтового сервера. |
| **Тип 5** (запрос на доступ к методическому обеспечению) | 1) принять транзакт;<br>2) поменять местами параметры узла-источника и узла-назначения;<br>3) размножить транзакт на некоторое случайное количество копий, варьируемое в заданном диапазоне;<br>4) осуществить маршрутизацию;<br>5) передать транзакт и копии модели передающего интерфейса сервера. |
| **Тип 6** (запрос на получение электронной почты) | 1) принять транзакт;<br>2) выбрать из модели почтового сервера все транзакты, у которых значение параметра узла-назначения совпадает со значением параметра узла-источника пришедшего транзакта;<br>3) удалить пришедший транзакт;<br>4) разбить выбранные письма на пакеты;<br>5) осуществить маршрутизацию копий;<br>6) передать выбранные транзакты модели передающего интерфейса сервера. |
| **Тип 8**<br>(сообщение в чат) | 1) принять транзакт;<br>2) размножить транзакт на количество копий, равный количеству конечных узлов;<br>3) присвоить параметру конечного узла каждой копии различные значения идентификаторов конечных узлов;<br>4) осуществить маршрутизацию для всех копий;<br>5) передать копии модели передающего интерфейса сервера. |
| **Тип 9** (обмен данными с узлом Internet) | 1) принять транзакт;<br>2) если в параметре узла-источника указан идентификатор конечного узла – присвоить параметру узла-назначения идентификатор узла Internet; если в параметре узла-источника указан идентификатор узла Internet – присвоить значение параметра адресата параметру узла-назначения;<br>3) осуществить маршрутизацию;<br>4) передать транзакт модели передающего интерфейса сервера. |
| **Тип 10** (сервис голосовой конференции)<br>**Тип 11** (сервис видео-конференции) | 1) принять транзакт;<br>2) если параметр направления равен 1 – удалить транзакт; если параметр направления равен 0 – переход к шагу 3;<br>3) размножить транзакт на количество копий, заданное значением параметра PConn;<br>4) инициализировать параметры узла-назначения копий значениями параметров с порядковыми номерами, равными порядковому номеру копии;<br>5) изменить параметр направления пришедшего транзакта на значение 1; поменять местами значения узла-источника и узла-назначения;<br>6) осуществить маршрутизацию транзакта и копий;<br>7) передать транзакт и копии передающему узлу. |
| **Тип 12** (размещение файла на сервере) | 1) принять транзакты одного семейства;<br>2) объединить транзакты одного семейства;<br>3) удалить полученный транзакт. |

Как отмечалось выше, сервер также является конечным узлом с особым назначением. Он не генерирует транзактов, однако также имеет передающий и принимающий физические интерфейсы. Структура моделей этих блоков в корне отличается от структуры моделей аналогичных блоков конечных узлов пользователей несмотря на то, что назначения их подобны. Так, из-за существенных различий в обработке транзактов различных типов, принимающий блок теперь берет на себя задачу разбиения на пакеты потоков, требующих этого. А вот модель передающего интерфейса упростилась по максимуму, и в модели сервера не содержит блоков инициализации параметров, а содержит только блоки, организующие приоритетное поступление транзактов в сеть, а также отвечающие за маршрутизацию и передачу транзакта в сеть (в этом плане блок передатчика сервера дублирует блоки передатчиков конечных станций пользователей).

## Заключение

В статье рассмотрены принципы построения и функционирования коммуникационных сетей, произведена декомпозиция сети дистанционного обучения, предложен подход к построению имитационных моделей по заданной топологии и характеристикам сети.

Решена проблема представления времени в модели для осуществления эффективного моделирования. Разработаны алгоритмы осуществления маршрутизации и моделирования конечных узлов сети, выделены основные составляющие элементы сети обучения, их взаимодействие и алгоритмы обработки передаваемых сообщений. Построена модель главного сервера, находящегося в вершине иерархической структуры сети дистанционного обучения. Произведены расчеты и даны рекомендации для трансляции характеристик сети в параметры модели. Предлагаемая модель может использоваться при принятии решений на построение коммуникационной сети дистанционного обучения.

Таким образом, полученные в статье результаты являются составной частью интеллектуальной системы проектирования информационных сетей дистанционного обучения.

## Литература

1. Ольшевский А.И. Интерактивная система проектирования топологии сетей дистанционного обучения на базе ДонГИИИ // Искусственный интеллект. – 2001. – № 2. – С. 52-56.
2. Ольшевский А.И. Интеллектуальная система проектирования информационных сетей дистанционного обучения на базе ДонГИИИ // Искусственный интеллект. – 2007. – № 1. – С. 244-249.
3. Нейлор Т.Х. Имитационные эксперименты с моделями экономических систем. – М.: Мир, 1975. – 334 с.
4. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. – СПб.: Питер, 2006. – 958 с.

***А.І. Ольшевський***
**Розробка алгоритмів генерації імітаційних моделей комунікаційних мереж дистанційного навчання**
Стаття присвячена питанням автоматизації імітаційного моделювання при проектуванні мереж дистанційного навчання. Для реалізації алгоритмів генерації імітаційних моделей на GPSS запропонована декомпозиція моделі комунікаційної мережі дистанційного навчання. Також приведені: представлення окремих частин моделі, алгоритми роботи і інтерпретація алгоритмів мовою моделювання в середовищі GPSS World.

*Статья поступила в редакцию 17.07.2008*